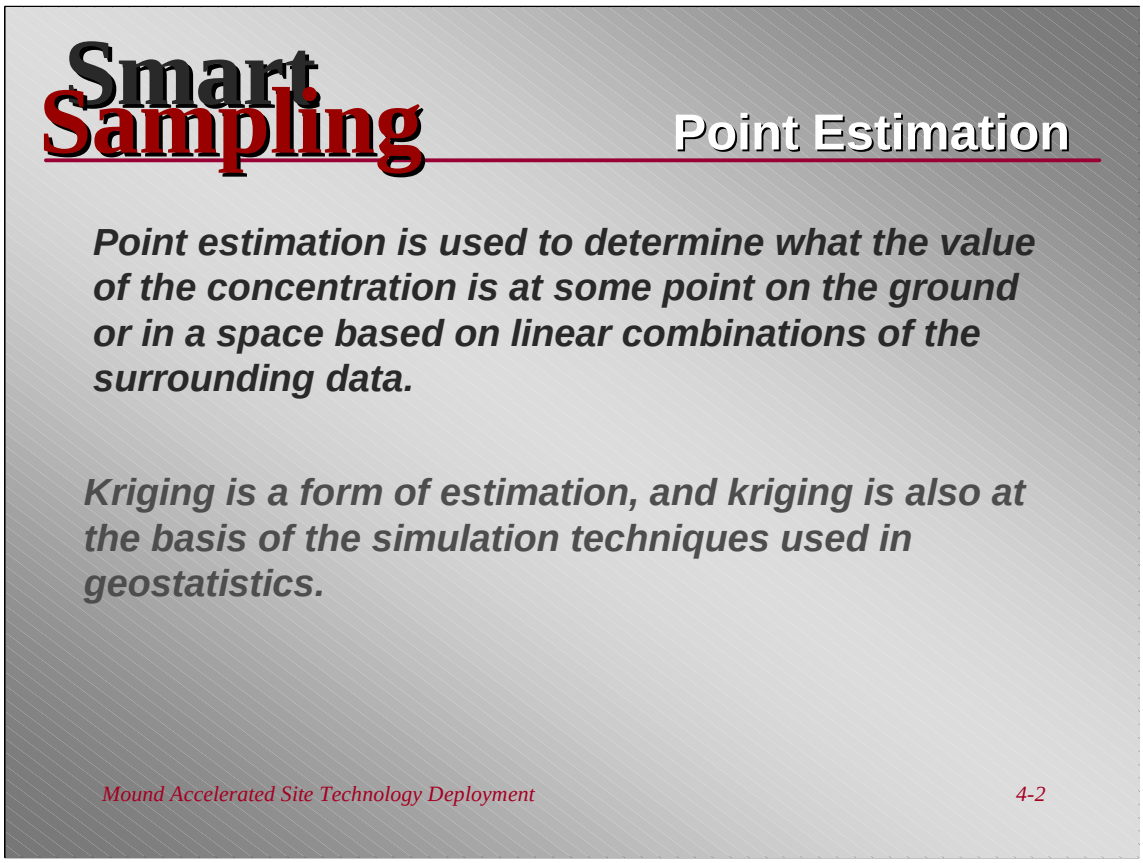
Smart Sampling

Section 4

Spatial Estimation



We have a means of modeling spatial correlation and we want to apply it to estimate values at unsampled locations. What we mean by point estimation is that at some point on the ground or in a space, we would like to determine the value of contaminate concentration.

The three techniques above are all data-driven, using some pre-existing data to interpolate to the surrounding locations.

Estimation is related to *kriging*, which lies at the basis of the simulation techniques used in geostatistics.

We will look at the three techniques above, as well as kriging. Other estimation techniques include *trend surfaces* and *splines*.

Smart Sampling

Estimation Techniques

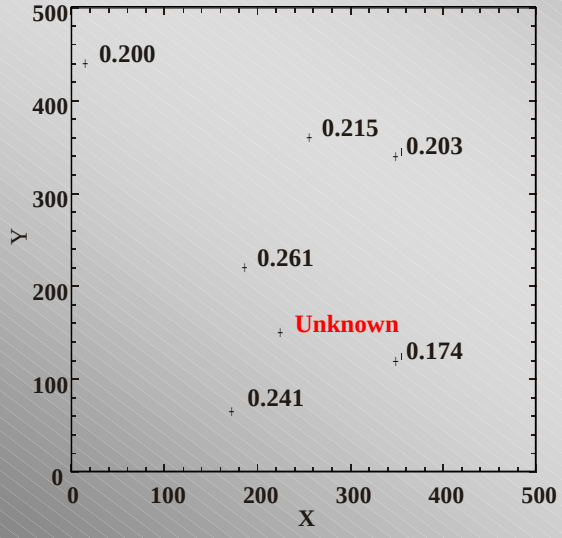
Example data-driven techniques (require some data to exist already) that interpolate to the surrounding locations:

- Nearest neighbor polygons
  (aka Theissen or Vornoi polygons)
- Local mean using surrounding data
- Inverse distance squared

We will look at the three techniques above, as well as kriging. There are other techniques such as: trend surfaces and splines.

Smart Sampling

Estimation Example Sample Data



- Porosity measured at 6 points.
- Want to estimate porosity at unknown point, x_0.

This map shows the locations of 6 samples of known porosity and one point where porosity is unknown. We will estimate the porosity at the unknown point using each of the three point estimation techniques.

Smart Sampling

Nearest Neighbor Polygons

- *Construct polygons around the samples that divide the space into regions*
- *Everywhere inside of the polygon is closer to the sample point enclosed by that polygon than to any other sample point*

Advantages:

- *Simple, fast, exact interpolator (at a point where the value is known, it returns that exact value)*

Disadvantages:

- *Discontinuities at polygon boundaries*
- *If not a lot of data and somewhat unevenly spaced, estimation dominated by the sparsely located points*

The idea is to Construct polygons that divide space into regions constructed such that every location within a polygon is closer to the sample location within that polygon than to any other sample location.

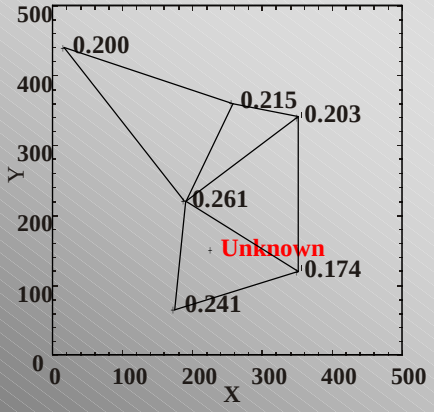
The advantage of a simple, fast, exact interpolator is that at a point where we know the sample value, this technique will return that exact value.

The disadvantage is that if there in not a lot of data and the data is somewhat unevenly spaced, the points that are further away from the others will dominate the estimation.

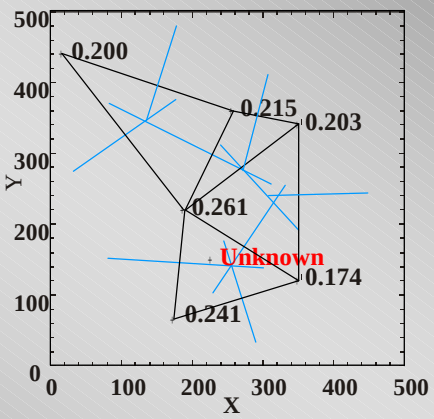
Smart Sampling

Nearest Neighbor Polygons

- *Connect each sample point to the neighboring sample points to create a series of triangles*



- *Draw a perpendicular bisector through each line.*



Estimated value at Unknown = .261

Around the point with value .261 there is a polygon forming, and every point within that polygon is closer to .261 than to any other sample point.

Every point within a polygon is assigned the value of the sample point in that polygon.

Estimated value at Unknown = .261

Smart Sampling

Local Mean Estimation

- *Use the mean of surrounding data as an estimate of the value at target location*

Advantages:

- *Simple, fast, few large errors (near the edges of the domain)*

Disadvantages:

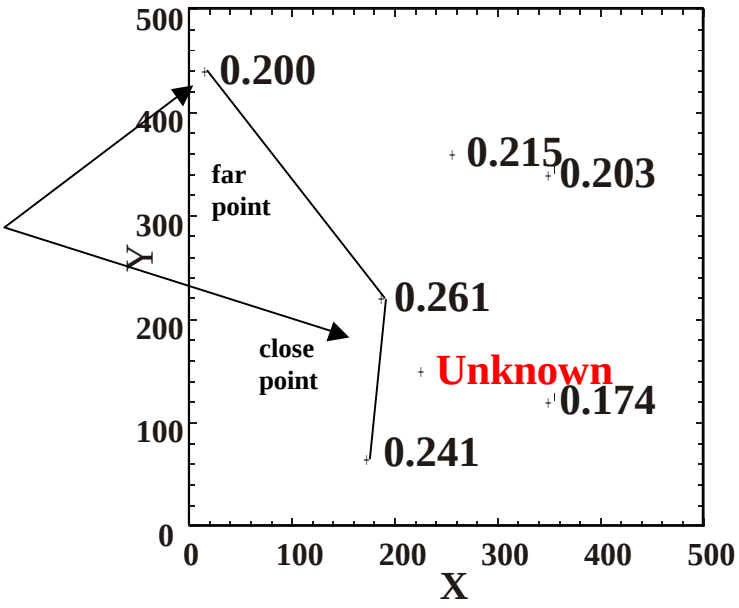
- *Not an exact interpolator (the average of the few surrounding data points won't necessarily return the exact value for a known point)*
- *Just what are the "surrounding data"?*
- *It has a smoothing effect on the data values. Any extreme values, high or low, will get smoothed out as they are averaged in with the surrounding values*

*For the example shown: mean = 1/n*SUM(data) = 0.216*

Should each surrounding point be weighted evenly?

Do some closer points influence the value at the unknown point more than some more distant points?

Should the more distant points be included in the average? If so, should they be given less weight?



For the example shown: mean = 1/n*SUM(data) = 0.216

Smart Sampling

Inverse Distance Estimation

- *Create weights for the data values that are inversely proportional to the distance from the unknown location, so the further away from the sample point, the smaller the weight*
- *The weighting function is the inverse distance raised to a power,* w

$$est = \frac{\sum_{i=1}^{n}\left[\left(\frac{1}{d_i^{?}}\right)z_i\right]}{\sum_{i=1}^{n}\frac{1}{d_i^{?}}}$$

Where:

d_i is the distance

z_i is the sample value

Advantages:

- *Simple, fast and includes distance in calculation of weights*

Disadvantages:

- *Not an exact interpolator*
- *As d goes to 0, the estimator "blows up".*

The Inverse Distance technique is similar to Local Mean, but the "surrounding" data are no longer equally weighted.

Each weight is normalized by sum of all weights and multiplied by its corresponding sample value

The most common value of ω is 2, this is called an inverse distance squared estimation (an option in many GIS programs).

In the equation, the denominator is the sum of all weights, the numerator is the weight times the respective value.

Smart Sampling

Inverse Distance Estimation

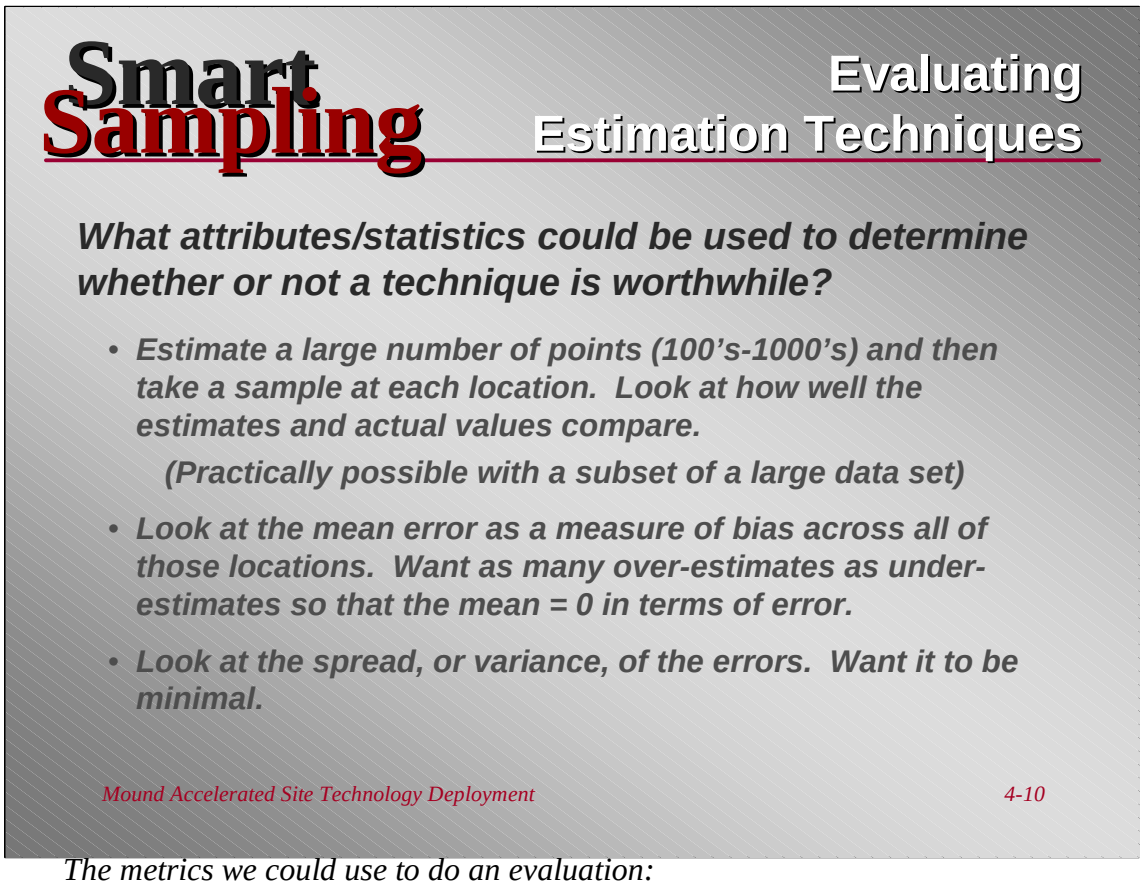
Inverse Distance Squared (w=2)

Estimation of Value at (235.0, 155.0)

Sample #	X	Y	Z Value	Distance from X0, Y0 (d)	Normalized Weight	Weighted Value
1	195	225	0.261	80.62	0.403	0.105
2	355	225	0.174	123.69	0.172	0.03
3	355	345	0.203	224.72	0.052	0.01
4	265	365	0.215	212.13	0.058	0.013
5	185	75	0.241	94.34	0.295	0.071
6	25	445	0.2	358.05	0.02	0.004

The final estimate is the sum of weighted values.

Estimate of value at Unknown = 0.233

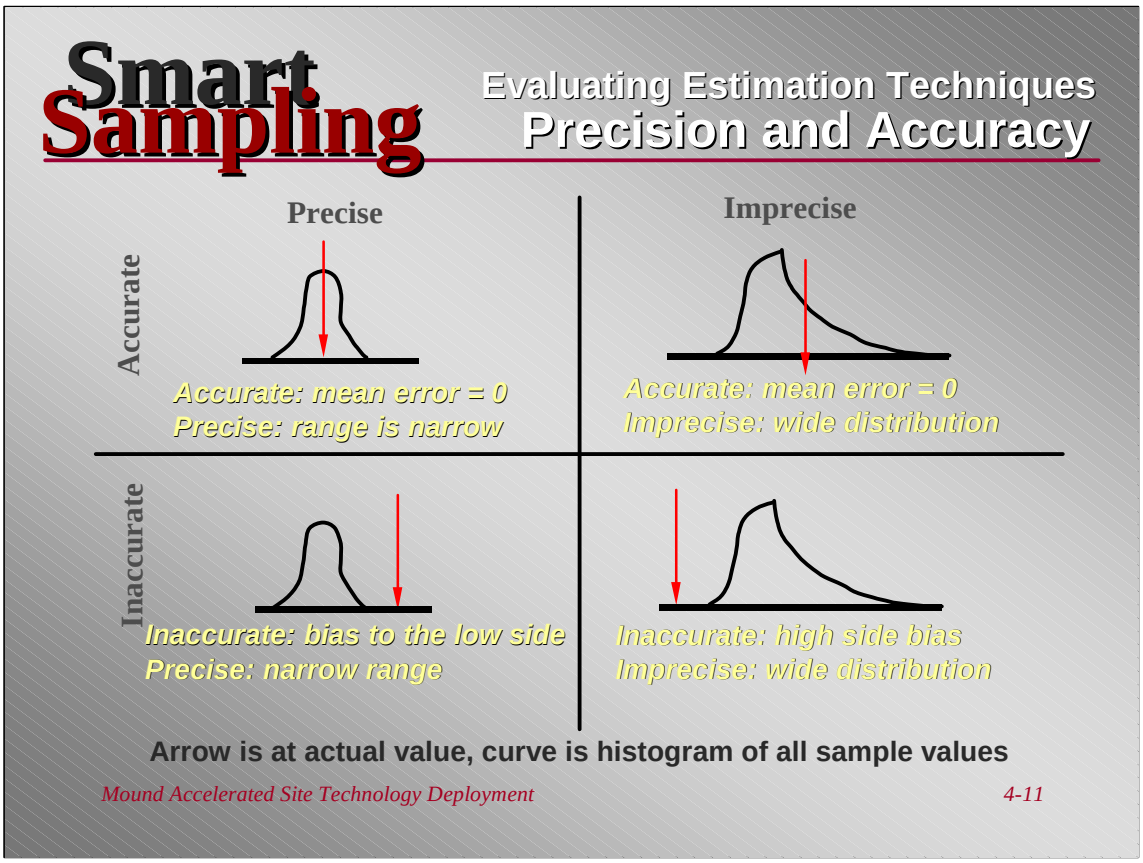


The metrics we could use to do an evaluation:

Confirmation Sampling / Jacknifing: We could estimate the values at the locations where we actually know the values and check to see how well a chosen technique performs (one at a time, pull each data point out of data set and estimate that data value using the surrounding data. Compare actual ("removed") value to the estimate made with surrounding data), **OR** imagine estimating a large number of points (100's-1000's) and then, after the estimation, taking a sample at each location. Look at how well the estimates and actual values compare. This is a "practical" possibility if you "hold out" a subset of a large data set when performing the estimation.

Mean Error: Look at mean error as a measure of bias across all of those locations. You want the bias to show consistent over- or under-estimation across all locations, so that the mean error = 0.

Spread/variance of Errors: Look at how widely spread the errors are. You want the variance to be minimal.

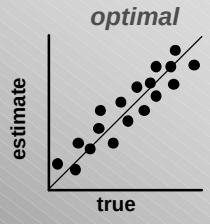


The Inaccurate/Precise model (lower left) is probably the worst case of all because though there is a precise, narrow range that lets you think you have small error, there is a strong bias to the low side. It suggests that the actual value is outside of the model.

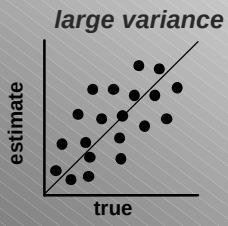
Smart Sampling

Evaluating Estimation Techniques
Scatter Plots

- *Build a model of the concentration at each point, the estimate.*
- *Take a sample and see how well the estimate and true values correlate.*



Optimal: The distribution is centered on the 45 degree line (accurate, unbiased) with a small spread.

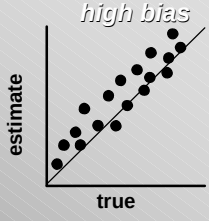


Large Variance: The distribution is accurate and unbiased, but the estimates are more variable causing a wider spread in the distribution (imprecise).

Smart Sampling

Evaluating Estimation Techniques
Scatter Plots

high bias

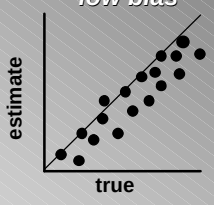


The distribution is precise but biased.

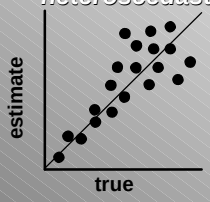
High bias: overestimates the true value

Low bias: underestimates the true value

low bias



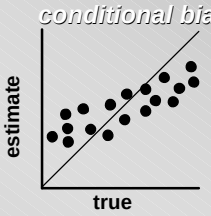
heteroscedastic



Heteroscedastic: The variance changes as a function of the value. So as the values increase, quality of the fit about the 45 degree line deteriorates.

Conditional: A small subsection appears to be optimal, but the low values tend to be overestimated and high values tend to be underestimated.

conditional bias

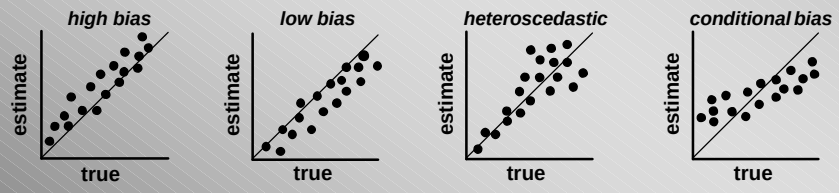


Smart Sampling

Evaluating Estimation Techniques
Jacknifing

Jacknifing:

- *One at a time, pull each data point out of data set and estimate that data value using the surrounding data*
- *Compare actual ("removed") value and estimate made with surrounding data on a scatter plot*



Smart Sampling

Kriging

Kriging is an estimator that uses a weighted linear combination of surrounding data to produce unbiased, minimum variance estimates.

Kriging weights are not based on Euclidean distance, but use the geometry defined by the variograms.

Ordinary Kriging (OK): ***Allows for local re-estimation of the global mean. The estimate is the sum of the product of the weights and the z values.***

$$\text{est} = \sum_{i=1}^{n} ?_i z_i$$

Simple Kriging (SK): ***Enforces the global mean on to each estimate. Sums the weighted residual of the estimate from the global mean and adds that sum to the global mean.***

$$\text{est} = \text{mean} + \sum_{i=1}^{n} ?_i [z_i - \text{mean}]$$

Mound Accelerated Site Technology Deployment *4-15*

Kriging is the basis of everything we do in geostatistical simulation.

Ordinary Kriging (OK) is similar to inverse distance squared. We have an estimate that is the sum of the product of the weights and the z values. The difference in this estimation is that the weights are not based on Euclidean distance but on the geometry defined by the variograms, generating a single best estimate at a location.

Simple Kriging (SK) uses residuals off the global mean to estimate residual of the global mean. Subtract the global mean from the estimate and multiply that by the weight, then add the sum of that to the global mean. This process is more precise if conditions present second order stationarity.

Theoretically, people like SK, but in practice OK is the way to go.

Smart Sampling

Kriging as a *BLUE*

Best Linear Unbiased Estimator (B.L.U.E.)

Does Ordinary Kriging (OK) fit the requirements of a B.L.U.E.?

Best:	***Tries to minimize the variance of the residuals and make the distribution precise***
Linear:	***Uses a weighted linear combination of the surrounding data***
Unbiased:	***Attempts to make the mean residual equal to zero***

Smart Sampling

Matrix Formulation

Calculation of Kriging Weights

Local covariance matrix that describes covariance between all samples in the local search neighborhood

$$\begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} & 1 \\ C_{21} & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} & 1 \\ C_{31} & C_{32} & C_{33} & C_{34} & C_{35} & C_{36} & 1 \\ C_{41} & C_{42} & C_{43} & C_{44} & C_{45} & C_{46} & 1 \\ C_{51} & C_{52} & C_{53} & C_{54} & C_{55} & C_{56} & 1 \\ C_{61} & C_{62} & C_{63} & C_{64} & C_{65} & C_{66} & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \\ w_6 \\ m \end{bmatrix} = \begin{bmatrix} C_{10} \\ C_{20} \\ C_{30} \\ C_{40} \\ C_{50} \\ C_{60} \\ 1 \end{bmatrix}$$

$\mathbf{C}$ $\quad \vec{w}$ $\quad \vec{D}$

Vector of covariances between each point in the search neighborhood and the location being estimated

To solve for vector of weights use matrix algebra: $W = C^{-1} * D$

Smart Sampling

Matrix Elements

C_{11} is the covariance at zero separation, the sill value

$$\begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} & 1 \\ C_{21} & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} & 1 \\ C_{31} & C_{32} & C_{33} & C_{34} & C_{35} & C_{36} & 1 \\ C_{41} & C_{42} & C_{43} & C_{44} & C_{45} & C_{46} & 1 \\ C_{51} & C_{52} & C_{53} & C_{54} & C_{55} & C_{56} & 1 \\ C_{61} & C_{62} & C_{63} & C_{64} & C_{65} & C_{66} & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \\ \omega_5 \\ \omega_6 \\ \mu \end{bmatrix} = \begin{bmatrix} C_{10} \\ C_{20} \\ C_{30} \\ C_{40} \\ C_{50} \\ C_{60} \\ 1 \end{bmatrix}$$

C_{61} is the covariance between points 6 and 1

Add an extra row and column to assure unbiasedness

Lagrange parameter, from adding a row for unbiasedness

The matrix of the covariances of all the sample points in the search neighborhood operates to take into account data redundancy.

Two points that are close to each other in one direction and have a high covariance are redundant, may want to weight them together as much as a single point in the opposite direction the same distance away.

After inverting C, which rescales the covariances, a large covariance becomes a small weight.

So the matrices take care of the clustering of the data points and the distance between unknown point and sample points in variogram space, not just Euclidean space. Put those two together to get the kriging weights.

Smart Sampling

Covariance Matrix

- *There is no guarantee of a unique solution to the matrix system. To ensure that there is only one unique solution, the system must be positive definite*
- *For estimates that are weighted linear combinations of other values, the variance about those estimates must be greater than or equal to zero*
- *Positive definite condition can be achieved by modeling variograms with positive definite functions, as long as one of the standard models is used (which are positive definite functions) the covariances will create a positive definite set of matrices*

SEAN: In my notes here it says to add a subsection on what we need to get started, data from site expected to be output using GIS software to put in correct format, mapping of data points right away…Is this a new slide or is it a new section??

Smart Sampling

Positive Definite

Guarantee that the variance of any random variable formed by a weighted linear combination of other random variables will be positive.

(Isaacs and Srivastava, 1989)

Smart Sampling

Unbiasedness

We want the mean, or expectation, of the errors to equal zero.

- Define error as: the estimated value – actual value

$$E(x_0) = \sum_{i=1}^{n} ?_i Z(x_i) - Z(x_0)$$

- Realize that the mean is stationary, so both the estimate and the actual value have the same mean.
- If the average error is set to zero, then:

$$\sum_{i=1}^{n} ?_i = 1.0$$

Sean - please note the intervening steps here...

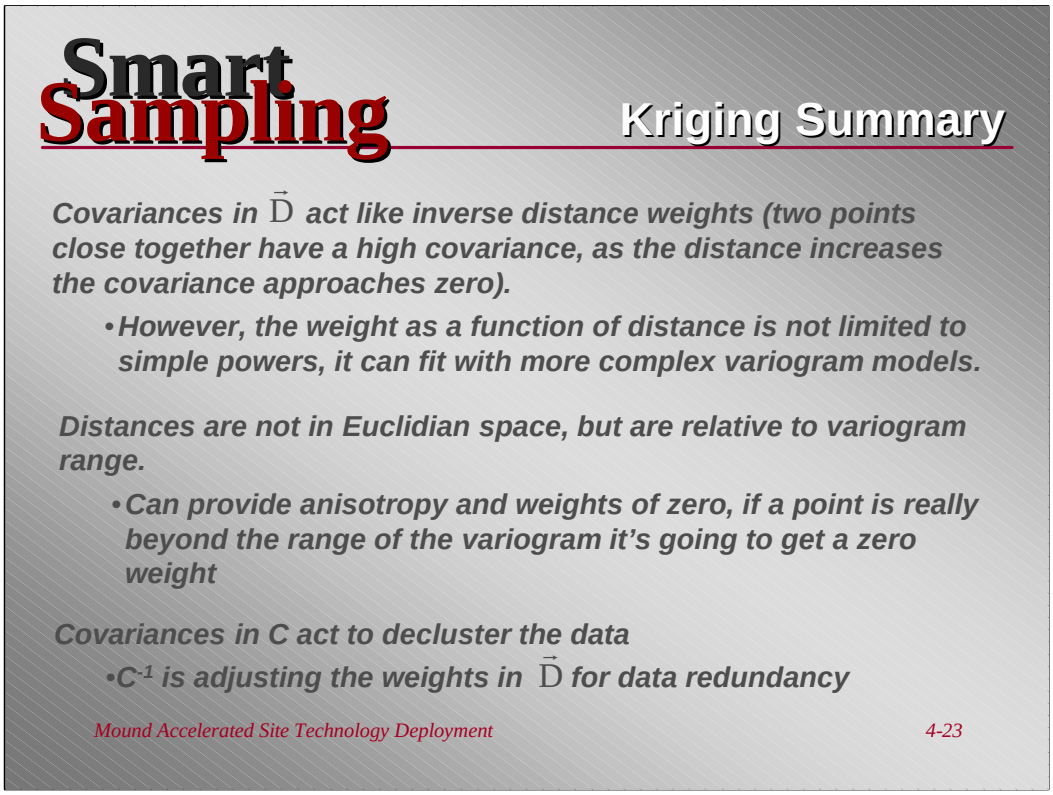
Smart Sampling

Lagrange Parameter

- **Lagrange parameter, μ, solves the problem of n+1 equations and only n unknowns created by the unbiasedness constraint**
- **Lagrange parameter is essentially another unknown**

$$\sum_{i=1}^{n} \omega_i = 1 \quad \blacktriangleright \sum_{i=1}^{n} \omega_i - 1 = 0 \quad \blacktriangleright 2\mu\left(\sum_{i=1}^{n} \omega_i - 1\right) = 0$$

In the Covariance matrix, we had six equations and six unknowns, then added the extra row and column for the unbiasedness constraint, which created seven equations and six unknowns. In order to fix that, use the Lagrange parameter.



The kriged weights leave more flexibility, can fit with more complex variogram models. Declustering of data is a function that other techniques do not provide.

Smart Sampling

Kriging Results

For the example problem, the normal-score variogram model is spherical, with a range of 125.0 (N-S) and 100.0 (E-W) and a nugget of 0.0

Location	Distance from X_0	Kriging Weight (radii = range)
1	80.6	0.541
2	123.7	0
3	224.7	0
4	212.1	0
5	94.3	0.459
6	358.1	0

Note that only points 1 and 5 fall within the search ellipse

Estimate of value at x_0 = 0.250

Mound Accelerated Site Technology Deployment 4-24

Note that only points 1 and 5 fall within the search ellipse.

The weights get multiplied by the value of the two points and the kriging estimate is the sum of those.

Kriging also provides an estimate of the error at a location; no other estimation technique does this.

Estimator results:

Nearest Neighbor Polygon : 0.261

Local Mean: 0.216

Inverse Distance: 0.233

Kriging: 0.250

Smart Sampling

Error Variance

- ***Kriging is unique among spatial estimation techniques in attempting to minimize errors, making the distribution tight.***
- ***Concisely, it is the variance of the errors that is minimized***

$$\sigma_e^2 = \frac{1}{n}\sum_{i=1}^{n}(e_i - \bar{e})^2$$

Where each error, e_i = the estimate - the true value
and $\bar{e}$ is the mean error

In this equation, we are looking at the difference between the error at any individual location and the mean error, squared.

Sean - my notes say you will make this smoother and add detail.

Smart Sampling

Error Variance

Short Version: ***The estimate of error variance is the total variance minus the weighted sum of covariances in $\vec{D}$ plus the Lagrange parameter***

Long Version: ***Derive a model of the error variance and minimize the modeled variance by setting partial derivative for each weighted covariance between datum and estimation location to zero.***

$$\hat{s}_e^2 = s_{Data}^2 - \frac{1}{n}\sum_{i=1}^{n}(?_i C_{i0}) + \mu$$

in matrix form:

$$\hat{s}_e^2 = s_{Data}^2 - (\vec{?} \cdot \vec{D})$$

where $\hat{\sigma}_e$ is the estimate

Mound Accelerated Site Technology Deployment *4-26*

$s_{Data}^2 - (\vec{?} \cdot \vec{D})$ expresses the fact that error variance in kriging is not a function of data values but rather of sample configuration - the proximity of other values.

For example: think of a simple system with a single data point in the search neighborhood, and we have a normalized covariance such that at zero separation distance the covariance is equal to 1. The variance of the data will be zero (one data point), the sum of the weighted covariances is 1, and μ is going to be zero. What we end up with in that system is that the estimate of the variances will be equal to zero.

Smart Sampling

Error Variance

- *Error variance is also called kriging variance or estimation variance*
- *Error variance is <u>not</u> a function of data values but only of sample configuration*
- *Error Variance is equal to zero at data location*
- *Like the kriging system, distribution of errors is <u>non-parametric</u>*

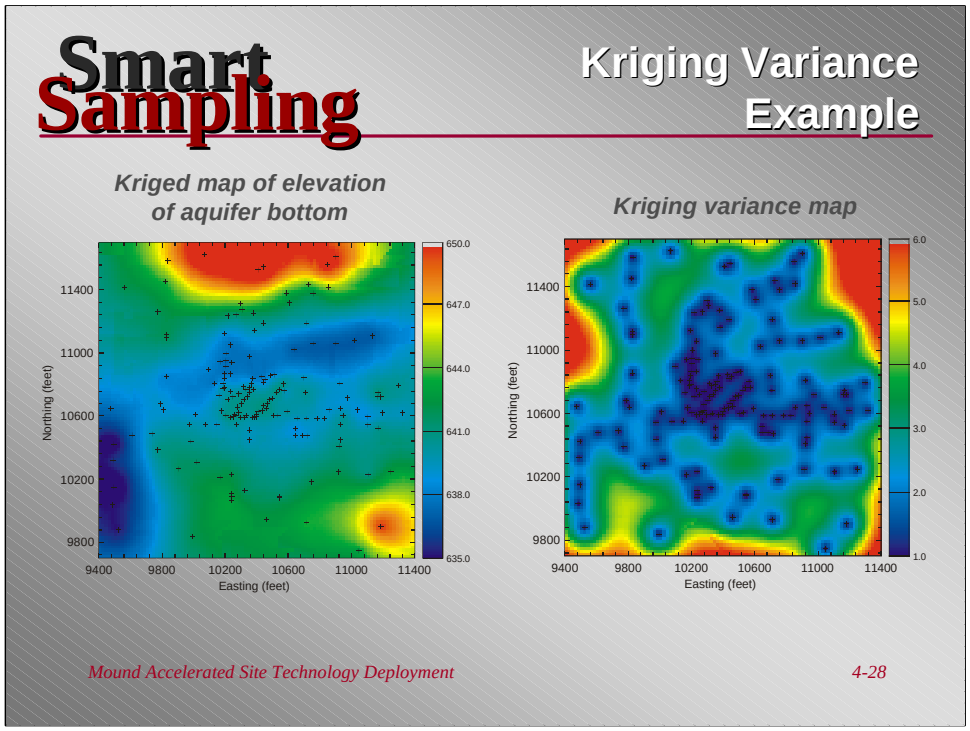
(although a gaussian distribution is often assumed for errors)

Recall: $$\hat{s}_e^2 = s_{Data}^2 - \frac{1}{n}\sum_{i=1}^{n}\left(?_i C_{i0}\right) + \mu$$

To minimize error variance, look only at the variance of the data and at the kriging weights.

It really comes down to the weights, which are a function of distance specified as the variograms and the variance of the data.

To locate additional samples in the area of greatest kriging variance, put the samples as far away as possible from any available data.

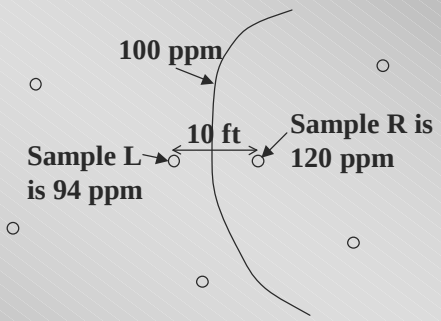


Kriging variance is not a function of data values, only of the proximity of other data points. It can be used as a means for locating additional samples - place the samples in the area of greatest kriging variance, as far as possible from any available data.

In this example, a lot of the sampling is focused on one area at the source of a leak, therefore there is low variance (blue zones). Additional samples in the area of high variance wouldn't give any information because they are outside of the area of regulatory concern. Need a criterium that is tied to the problem, not just to the spatial extent of the data.

Smart Sampling

Kriging Variance Example



Since Sample R and Sample L are very close to each other, there will be a small error variance, but if the regulatory threshold is 100 ppm, then R fails and L passes.

Example: Two samples, 10 feet away from each other in a very large domain. Sample R value is 120 ppm, Sample L is 94 ppm. There are some other samples in the area. Since R and L are very close to each other, there will be a small error variance, but if the regulatory threshold is 100 ppm, then R fails and L passes. There is significant uncertianty as to where that 100 ppm line actually falls. So even though the kriging variance tells us that it's a very small error, if we're interested in sampling based on an action level that falls between these samples, the uncertainty directs us to target our sampling in this location.

Kriging variance is not a function of the data values at all, only of the proximity of other data points.

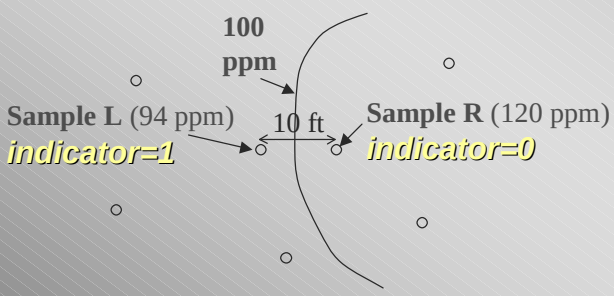
At any location, to determine if we want to take another sample:

- Get the best estimate from kriging
- Get the variability at the same location from the kriging variance map, we assume the shape is Gaussian.

Smart Sampling

Indicator Kriging

- *Indicator kriging is basically kriging on an indicator (binary) transform of the of the data.*
- *Transforming data to zeros and ones using a threshold.*
- *Values less than or equal to the threshold are given a value of 1, and values above that are given 0's.*



Indicator Kriging

- Transform data to zeros and ones using a threshold.
- Krige data using indicator variogram (Kriging the indicators requires the variogram to be modeled in the indicator space)
- Resulting map shows p(exceed) threshold (all values in [0,1])

Smart Sampling

Map of Indicator Kriged Probability of Exceedence

Sean - Make Map

Smart Sampling

Practical Assessment of Model

Goal: *try to test technique under conditions as close as possible to true estimation conditions*

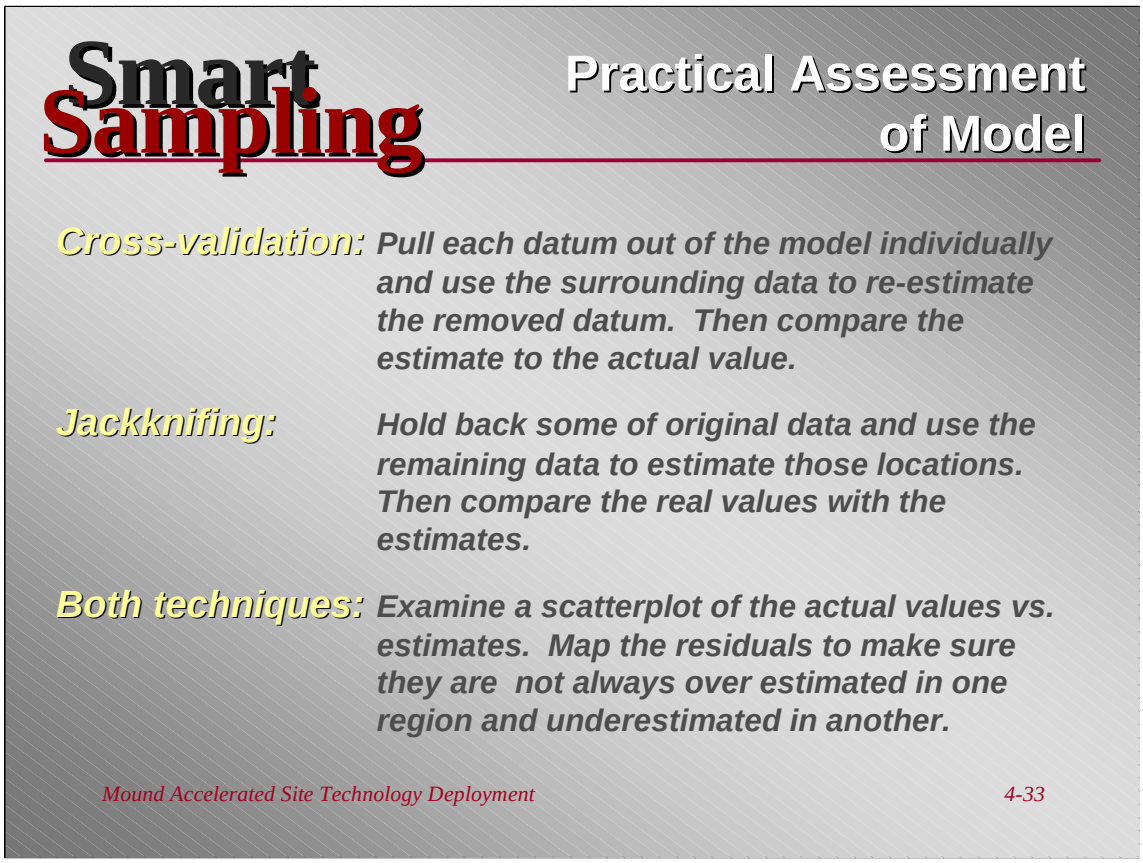
Numerically (without doing samples), there are two possible techniques:

- *Cross-Validation*
- *Jackknifing*

"True estimation conditions" refers to the regulator that will come in and sample a spot that is said to be clean to see if it is.

Cross-validation and Jackknifing are two numerical techniques that we can perform to test if our model demonstrates the contaminant distribution at all points without going out into the field for more samples.

Residuals can show if you are imbalanced between high and low errors.



Cross-validation: If we have a lot of data points, we can go around one at a time and pull out and re-estimate each point and see how close the estimate is. Afterwards we can make a scatterplot and plot actual vs. estimate to see how close we are, we can also map those residuals to make sure we are not always over estimating in one region and underestimating in another.

Jackknifing: If we had 1000 samples, we could estimate the concentration across the site using only 800 of the samples, then we could add in those 200 additional locations where we held the data back and compare the real values with the estimates.